

// laboratório codex 05 · orçamento 4.0

python + ia + app

>_ Laboratório Codex 05

O orçamento inteligente, com IA

Banco de dados, previsão de vendas, mix ótimo de produção, painel orçado x realizado, alertas e relatório executivo: o orçamento que ajuda a decidir.

// apostila do aluno V · o projeto em etapas

01 Banco de dados

02 Previsão de vendas

03 Mix ótimo

04 Orçado x realizado

05 Alertas

06 Relatório executivo

07 Preços-sombra

08 Apresentação



Prof. Caio Haffner

Mestre em Desenvolvimento Econômico · Unicamp

1 Como usar esta apostila

Esta é a **Apostila V de Gerenciamento Orçamentário**, a etapa avançada da trilha: ela continua a **Apostila IV** (o App do Orçamento da Mesas Ipê no ar (Apostila IV)). Agora você vai dar **inteligência** ao App do Orçamento da Mesas Ipê: um banco de dados com o histórico, a previsão das vendas, o melhor mix de produção, um painel do orçado x realizado, alertas automáticos e um relatório executivo para a diretoria.

1. Guardar

Banco de dados

2. Prever

Mesas vendidas
por mês

3. Otimizar

Mix de produção

4. Vigiar

Alertas

5. Decidir

Relatório
executivo



O PROJETO: O ORÇAMENTO INTELIGENTE

Nas Apostilas III e IV, o app monta o orçamento. Agora ele passa a ajudar a decidir: prevê as vendas, calcula quantas mesas e cadeiras produzir para lucrar mais, avisa quando o realizado foge do orçado e escreve o relatório do mês.

Ao final, você deverá conseguir

Habilidade	Como saber que consegui	Ok
Usar um banco de dados	O histórico fica num banco SQLite e o app consulta com SQL.	☐
Prever	Comparo média móvel e tendência e meço o erro.	☐
Otimizar a produção	Calculo o mix de mesas e cadeiras que dá o maior lucro.	☐
Mostrar num painel	Os resultados aparecem num painel interativo.	☐
Criar alertas	O app avisa quando um número sai do padrão.	☐
Relatar	O app gera o relatório executivo do mês.	☐

1

REGRA DE OURO

Você é responsável pelo código que entrega, mesmo quando a IA o escreveu. **Não entregue código que você não consegue explicar.** Aqui vale mais uma: **previsão, simulação e otimização são apoio à decisão.** Quem decide continua sendo uma pessoa, que entende os números.



LEIA TUDO, NA ORDEM

Leia esta apostila **desde a primeira palavra** e faça tudo o que ela manda, na ordem. Cada etapa usa o que a anterior construiu.

A Apostila V começa onde a IV termina: **o App do Orçamento da Mesas Ipê no ar, no GitHub, com testes e dados reais**. Se ainda não está assim, termine a IV antes.

Instale as bibliotecas novas (uma vez só)

TERMINAL

```
python -m pip install numpy scipy plotly fpdf2
```

Biblioteca	Para que serve
sqlite3	Banco de dados num arquivo (já vem com o Python).
numpy e scipy	Contas, simulação e otimização (a melhor decisão dentro dos limites).
plotly	Gráficos interativos dentro do app: zoom, filtro e valor ao passar o mouse.
fpdf2	Gerar o relatório executivo em PDF.

Ao longo das etapas, a pasta do projeto vai ganhar:

TERMINAL

```
Laboratorio_Codex_G03/  
├─ dados/historico.db ← novo: banco de dados  
├─ calculos.py  
├─ previsao.py        ← novo  
├─ otimizacao.py      ← novo  
├─ painel.py          ← novo  
├─ alertas.py         ← novo  
├─ app.py  
└─ tests/
```



LEMBRETES DAS APOSTILAS ANTERIORES

Peça sempre um **plano** antes do código · teste com contas feitas por você · peça o **diagnóstico antes da correção** · nada vai para a internet sem passar nos testes · nunca coloque senhas no código.

3 Etapa 1 • Guardar: o banco de dados

Planilhas soltas não aguentam anos de histórico. Um **banco de dados** guarda tudo e responde perguntas em segundos.

PROMPT 1

Crie dados/historico.db (SQLite, que já vem com o Python) com as tabelas vendas (mês, produto, quantidade, preço), custos (mês, item, orçado, realizado) e caixa (mês, entradas, saídas), importando os dados das Apostilas III e IV. Crie banco.py com funções que usem SQL para:

1. a receita por trimestre;
2. o custo por mesa orçado e o realizado, mês a mês;
3. o saldo de caixa no fim de cada trimestre.

Antes de criar, explique em 5 linhas o que são tabela, linha, coluna e consulta SQL.



CHECKPOINT

A receita e o saldo de caixa batem com o orçamento da Apostila III para o mesmo trimestre.



LEIA O SQL

Peça ao Codex: "explique esta consulta SQL linha por linha". SELECT escolhe as colunas, WHERE filtra as linhas, GROUP BY agrupa e ORDER BY ordena.

4 Etapa 2 · Prever: o que vem por aí?

Para decidir o próximo mês, é preciso uma **previsão**. Duas formas simples e honestas: média móvel e tendência.



média móvel de 3 períodos = média dos 3 últimos ·
tendência: valor = $a + b \times \text{período}$ (regressão)

Erro absoluto médio (EAM) = média de $|\text{real} - \text{previsto}|$ nos períodos em que dá para testar



MESAS VENDIDAS POR MÊS (12 MESES)

400 · 420 · 410 · 430 · 450 · 440 · 470 · 480 · 475 · 500 · 510 · 505

PROMPT 2

Crie `previsao.py` com:

1. `media_movel(serie, n=3)` e a previsão do período 13;
2. `tendencia(serie)`: a e b da reta por mínimos quadrados e a previsão dos períodos 13 e 14;
3. o erro absoluto médio da média móvel nos períodos 4 a 12.

No app, mostre um gráfico com o histórico, as duas previsões e uma frase dizendo qual erra menos.

Resultado	Esperado	Calculado
Média móvel (período 13)	505,0	
Tendência: a e b	$a \approx 390,7$ · $b \approx 10,28$	
Tendência (períodos 13 e 14)	524,3 e 534,6	
Erro absoluto médio da média móvel	20,9	



PARA PENSAR

A previsão das vendas é o primeiro número do orçamento: tudo depende dela. O que acontece com o caixa se a previsão errar 13 meses para mais?

5 Etapa 3 · Otimizar: o mix de produção

A fábrica tem horas limitadas. Quantas **mesas** e quantas **cadeiras** produzir para lucrar mais? É um problema de **programação linear**.

★ A FÁBRICA NO MÊS

Lucro: mesa R\$ 120, cadeira R\$ 50 · Marcenaria: mesa 4 h, cadeira 1 h, até 2.000 h · Acabamento: mesa 2 h, cadeira 1 h, até 1.200 h · Cadeiras: no máximo 900 (limite de vendas).

PROMPT 3

Crie `otimizacao.py` com `scipy.optimize.linprog`:

1. o mix de mesas e cadeiras que dá o maior lucro, respeitando as horas e o limite de cadeiras;
 2. as horas usadas e as que sobram em cada setor;
 3. uma conferência: o lucro de 3 outros mixes possíveis, todos menores que o ótimo.
- Antes do código, escreva o problema em português: o que decidir, o que maximizar e os limites.

Resultado	Esperado	Calculado
Mesas e cadeiras no mix ótimo	400 mesas e 400 cadeiras	
Lucro do mix ótimo	R\$ 68.000	
Horas que sobram (marcenaria e acabamento)	0 h e 0 h	

i PARA PENSAR

As duas áreas usam todas as horas. Se a fábrica pudesse contratar horas extras em só um setor, qual escolheria? (O desafio responde com números.)

Etapa 4 · Mostrar: o painel do orçado x realizado

A diretoria quer ver, num relance, onde o realizado fugiu do orçado.

PROMPT 4

Acrescente ao app a aba "Orçado x realizado" com plotly: barras do orçado e do realizado por mês para receita, custos e caixa, a diferença em % com cor (verde até 5%, amarela de 5% a 10%, vermelha acima de 10%) e um filtro por trimestre. Ao passar o mouse, mostre os valores e a diferença.



CHECKPOINT

O painel abre no app, o filtro de trimestre funciona e as diferenças acima de 10% aparecem em vermelho.

Etapa 5 · Vigiar: alertas automáticos

Ninguém olha todos os números todos os dias. Os **alertas** mostram só o que pede atenção.

PROMPT 5

Crie `alertas.py` com regras que o usuário pode ajustar na barra lateral:

1. caixa abaixo do saldo mínimo definido pela diretoria;
2. custo por mesa 5% acima do orçado;
3. vendas do mês 10% abaixo da previsão;
4. previsão de vendas acima da capacidade da fábrica (horas disponíveis).

No topo do app, mostre os alertas em vermelho (crítico) e amarelo (atenção), do mais grave para o menos grave.

Alerta	Quantos apareceram	Faz sentido?
Caixa abaixo do mínimo		
Custo por mesa acima do orçado		
Vendas abaixo da previsão		
Previsão acima da capacidade		



ALERTA DEMAIS É ALERTA NENHUM

Se aparecerem 30 alertas, ninguém olha mais. Ajuste as regras para mostrar só o que exige decisão.

Etapa 6 · Decidir: o relatório executivo

Quem decide não quer tabelas: quer saber o que aconteceu, o que vem aí e o que fazer.

PROMPT 6

Instale fpdf2 (python -m pip install fpdf2) e acrescente o botão "Relatório executivo (PDF)",
de uma página: o resultado do mês (receita, custo por mesa, lucro e caixa), a
previsão de vendas dos próximos 2 meses,
o mix ótimo de produção, os 3 alertas mais graves e 3 recomendações escritas pelo
app,
cada uma com o número que a justifica. Atualize os testes e publique a nova versão
(como na Apostila IV).



CHECKPOINT

O PDF tem uma página, os números batem com o app e a nova versão está no ar pelo link da Apostila IV.

9

Desafio extra · Preços-sombra: onde vale a hora extra

O **preço-sombra** diz quanto o lucro sobe com uma hora a mais em cada setor. É a resposta para "onde investir".

PROMPT 7

desafio

Use os resultados do linprog (res.ineqlin.marginals, com o método "highs") para mostrar quanto o lucro sobe com 1 hora a mais de marcenaria, de acabamento e com 1 cadeira a mais no limite de vendas. Confira refazendo a otimização com 10 horas a mais em cada setor. Se a hora extra custa R\$ 30, em qual setor vale pagar? Explique o resultado em 3 linhas antes de mostrar o código.



CHECKPOINT

Marcenaria vale R\$ 10 por hora, acabamento R\$ 40 por hora e o limite de cadeiras R\$ 0 (sobram 500). Hora extra a R\$ 30 só compensa no acabamento.

10

Use o Codex como revisor do projeto

REVISAR A ORGANIZAÇÃO

Revise o meu projeto como um programador experiente revisaria o de um iniciante. Não faça alterações. Aponte: funções muito longas, código repetido, nomes pouco claros e cálculos misturados com a tela. Separe o que é obrigatório do que é opcional.

REVISAR O CONTEÚDO

Revise os cálculos do meu projeto como um professor de Gerenciamento Orçamentário. Não faça alterações. Confira o SQL, a previsão (e o erro medido), os limites da programação linear e as regras dos alertas e liste o que pode estar errado.



GRÁFICO BONITO NÃO É GRÁFICO CERTO

Confira pelo menos um número de cada gráfico com a calculadora ou com a Apostila II. A IA erra eixos, unidades e somas com frequência.

Entrega: apresente o seu projeto

Um projeto só existe quando alguém o vê funcionando. Prepare uma **apresentação de 3 minutos** e um registro para o seu portfólio.

Roteiro de 3 minutos

Tempo	O que mostrar
30 s	O problema que o app resolve e para quem.
1 min 30 s	O app funcionando: mexa nos controles e mostre o mix ótimo de produção e o painel do orçado × realizado.
30 s	Um erro que a IA cometeu e como você descobriu.
30 s	O que você faria a seguir, se tivesse mais tempo.

O que entregar

- ☐ A pasta do projeto, com todos os arquivos .py e os dados
- ☐ O relatório em Excel gerado pelo app
- ☐ Três prints do app (telas principais)
- ☐ Um arquivo LEIAME.md explicando o que o app faz e como rodar (peça um rascunho ao Codex e revise)
- ☐ O diário de prompts: pelo menos 5 prompts importantes e o que aconteceu



PARA O SEU PORTFÓLIO

Um post no LinkedIn com um vídeo curto do app funcionando ("Construí em Python, com ajuda de IA, um app de orçamento com otimização e previsão") mostra a recrutadores algo que poucos estudantes têm: **uma ferramenta real feita por você.**

Reflexão individual

1. Qual etapa foi a mais difícil, e como você a resolveu?

2. Em que momento a IA errou, e como você percebeu?

3. O que você aprendeu de Gerenciamento Orçamentário usando previsão e otimização que não tinha aprendido nas apostilas anteriores?

12 Como você será avaliado

Critério	0	1	2	3
O app funciona	Não abre	Abre com erros	Funciona	Funciona e trata entradas inválidas
Conteúdo	Cálculos errados	Alguns certos	Quase todos certos	Todos certos e conferidos
Gráficos	Nenhum	Sem títulos ou eixos	Corretos	Corretos e fáceis de ler
Relatório	Não gera	Gera incompleto	Gera completo	Completo e organizado em abas
Uso do Codex	Não usou	Só copiou	Usou com compreensão	Usou de forma crítica
Apresentação	Não apresentou	Confusa	Clara	Clara, no tempo e com o erro da IA

Total máximo: 18 pontos, convertidos para a escala da disciplina.

13 Consulta rápida

Comandos

TERMINAL

```
python -m pip install numpy scipy
plotly fpdf2
python -m pytest                # roda
os testes
python -m streamlit run app.py  # abre
o app
git add . ; git commit -m "..."; git
push # publica
```

Streamlit

<code>st.title · st.header</code>	Títulos
<code>st.number_input · st.slider</code>	Entradas de número
<code>st.selectbox · st.file_uploader</code>	Escolha e upload
<code>st.metric · st.dataframe</code>	Indicador e tabela
<code>st.pyplot(fig)</code>	Mostrar um gráfico
<code>st.tabs · st.sidebar</code>	Abas e barra lateral
<code>st.download_button</code>	Botão para baixar o Excel

Glossário

Biblioteca

Código pronto que você instala para ganhar recursos (gráficos, telas).

pip

O instalador de bibliotecas do Python.

DataFrame

Tabela do pandas, com linhas e colunas.

App web

Programa que roda no navegador.

SQLite

Banco de dados num arquivo só, sem servidor.

SQL

Linguagem para consultar tabelas (SELECT, WHERE, GROUP BY).

Média móvel

Média dos últimos períodos, usada para prever o próximo.

Erro absoluto médio

Quanto a previsão erra, em média, para mais ou para menos.

Programação linear

Método que acha a melhor decisão respeitando limites (horas, materiais).

Preço-sombra

Quanto o lucro sobe com uma unidade a mais de um recurso escasso.

1

LEMBRE-SE

A IA é uma ferramenta. Conhecimento, julgamento, validação e responsabilidade continuam sendo de quem programa. Nada nesta apostila é recomendação profissional: é aprendizado.